Chapter 7

Computer Software

7.1   Introduction

In previous chapters, we discussed about different parts and configurations of the computer. It has been mentioned that programs or instructions are given to the computer to do a specific task. So it is necessary to provide sequence of instructions so that your work can be done. **Software** is a single/collection of programs that performs a particular task. A computer is useless without software and vice versa. In this chapter, you will learn about different types of commonly used software and their functions. There are two types of software on the basis of functionality: *System Software* and *Application Software*.

The **Application Software** is a program created to perform a specific task for a user. For example, to create a document, a word processing software is used such as Microsoft Word. Whereas, to create spreadsheets, we can use Excel or Lotus 123, etc.

7.2   System Software

The software used to control, monitor, or facilitate the use of the computer is called **System Software**.

System Software provides the interface to the computer and controls basic operations like saving data on the disk, making computer to work for our use, printing a document, etc. The system software includes the operating system, the language translators, the linkers, loaders and other utility programs, etc.
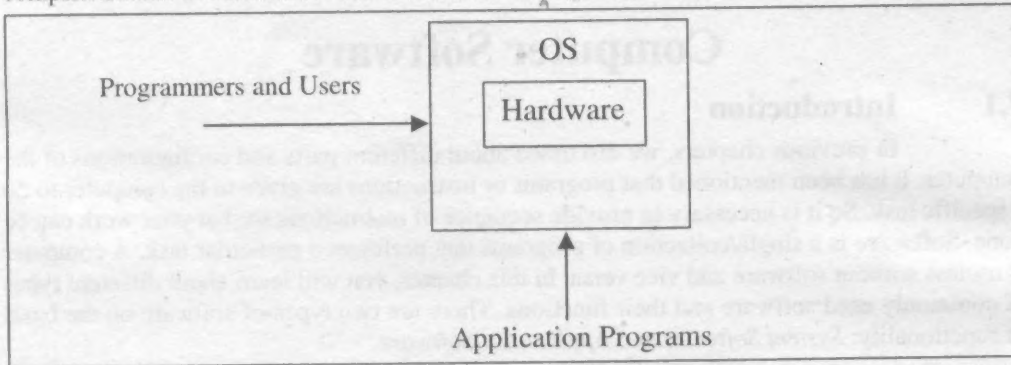
Operating System

In order to use computer, to solve some problem, a user has to write instructions (program). But other than writing instructions for solving the problem every programmer will also have to write instructions for the following tasks:

- Read data from the input devices.
- Show results on the output devices.
- Perform memory management i.e, load programs and data into memory and manage memory.
- Organize data on the storage devices.

These tasks are very complex and only expert programmers can write these instructions. To make the use of computer easy, these programs are provided in the form of a software, known as Operating System (OS).

An operating system is a set of programs running on a computer system, and provide an environment in which other programs can be executed and the computer system can be used effectively.

The operating system provides the most common functions needed by any user. In this way the operating system wraps around the hardware and saves its users from the complex details of hardware. This is shown in the figure below.

From this figure it is obvious that the OS will not only provide programs for doing different tasks but will also provide an interface to its users (i.e. programs, programmers etc).

Types of Operating System

Depending upon the working, the Operating Systems are divided into the following main types:

- Batch Processing Operating Systems.
- Real-Time Operating Systems.
- Single User Operating Systems.
- Multi-User Time Sharing Operating Systems.
- Network Operating Systems

7.3   Functions of an Operating Systems

Following are the main functions performed by the most Operating Systems:

- Manage Hardware Resources
- Load and Execute programs
- Memory Management
- Secondary Memory Store Management
- Providing interface to the users

There are two commonly used user interfaces:

Command Line Interface: In such interfaces the users communicate with the Operating System by typing commands using a keyboard. Each command given to the OS activates one of the many programs in the OS. Example of such an interface is the command prompt provided by MS-DOS to its users.

Graphical User Interface (GUI): The GUI interface consists of Window, Menus, Icons and pointers. The user of the system communicates with the Operating System by selecting commands from the menus or by selecting different icons by using a pointing device like mouse. MS-Windows is a well-known example of an Operating System with a GUI interface.

In MS-Windows the user selects commands by using a mouse and a keyboard e.g. Windows XP, Linux etc.

7.4 Language Translators

This is another very important category of system software and has played a very important role in the development of general-purpose computers. There are three main types of language translators ie, assemblers, compilers and interpreters.

7.4.1 Assembler

Assembler is a program that converts an assembly language program into machine language.

In the early days of computers, the programs had to be written in machine code instructions using binary codes. So writing programs was a very difficult and time-consuming task. It was also prone to errors and removing these errors was another difficult task.

Assembly language was developed to make the job easier. Assembly language allowed the use of symbolic codes for machine instructions; these codes are called mnemonics (pronounced as Ne-Monics). Obviously, writing programs in Assembly language is much easier than writing instructions in binary. An Assembly language provides one line of code for each machine instruction.

7.4.2 Compiler

A **compiler** is a program that translates a source program (written in some high-level programming language) into machine code as a whole.

The generated machine code can be later executed. A compiler first reads the whole program before executing it. The errors in the code are pointed out and then the machine language code is generated. This output is known as **object program**. The object program executes very fast as it is directly understandable by the computer. Today most of the languages use compilers. Once a program has been translated into machine code it can be loaded into the machine and can be executed. This process of translation is shown in the figure below.
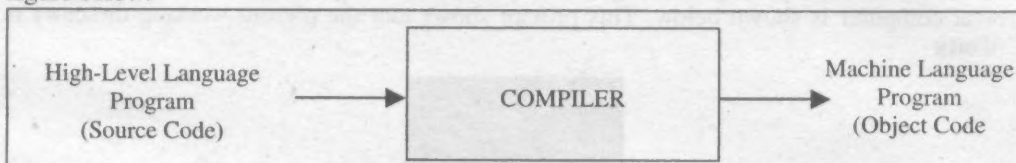
As shown in the above figure, the high level program is called the **source code** and the translated program is called the **object code**

7.4.3 Interpreter

An **interpreter** looks at each line of the source program, decides what that line means, checks it for possible errors and then executes that line.

If one of the lines is executed repeatedly, it must be scanned and analyzed each time, greatly slowing down the solution of the problem at hand. Thus interpreter executes code line by line. If an error is encountered, the execution is stopped at that line and user is notified of the line where the error occurred. After the error has been removed, the user has to run the whole program from the beginning. This whole process slows down the performance of the interpreter as compared to the compiler. Most of the languages used for writing short scripts of code use interpreters.

7.5 Disk Operating System (DOS)

It is a single user operating system and has been very popular on microcomputers up to mid 1990s. DOS was designed by IBM (International Buisness Machines). DOS resides on

disk and controls the overall functioning of the computer. It performs the following major tasks:

- Control input and output devices
- Execute user programs
- Manage system resources
- Provide user interface
- Memory management

DOS does not provide networking features. To connect a computer running on DOS, some third party networking software should be installed.

7.5.1 DOS Files

There are three important files in DOS which it uses for different purposes. These are **Batch Files, Command Files** and **Executable Files.** These files are identified by their extensions which are .bat, .com, and .exe respectively.

In **Batch Files**, one or more commands are grouped together. The name of batch file acts as a command for the DOS. When the name of file is given to DOS on command prompt, the commands written in the batch file are executed sequentially without involving user to type every command on command prompt.

Executable Files are in executable form i.e. these are ready to run on the computer. These files contain instructions in machine language whereas **Command Files** contain DOS commands.

7.5.2 User Interface of DOS

DOS provides a command driven interface to the users. In a command driven interface the users can use the keyboard to type the command on the command prompt. Mostly the DOS prompt shows the present working directory. DOS interprets all commands with reference to the present working directory until explicitly stated. The DOS prompt on a typical computer is shown below. This prompt shows that the present working directory is **C:\Data**
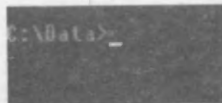
In Desktop computers Microsoft Windows is the most commonly used operating system and DOS is no more in common use. Windows also provide a command prompt similar to the DOS prompt and provide all commands used in DOS. Actually **Windows 9.X** operating system's work on top of DOS **Windows NT** and **Windows 2000** also provides its users with a command prompt and commands used in DOS. So learning the DOS commands can be very useful.

If you have DOS installed as the main Operating System then the DOS prompt is automatically shown when you switch on your computer. If you have **Windows 98** installed on your machine, then you can go to the DOS prompt as shown in the above figure and explained below.
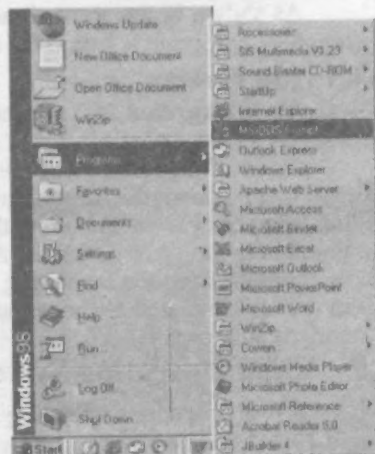
To activate the DOS prompt, first click at the start button shown on the desktop of your computer. Then choose the Programs menu by using the mouse or the keyboard. From this menu choose MS DOS prompt option. You will see the following Screen.

To expand the prompt to full screen, press the **ALT + Enter** keys. Note that you can restore to the original screen by pressing these keys again. It is also important to note that you can close this screen by entering the **Exit** command on the prompt.

If you have **Windows NT** or **Windows 2000** installed on your computer, then you can use a similar procedure to locate the command prompt.
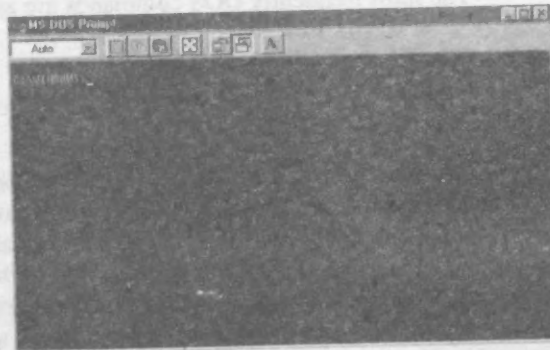
7.6   DOS Commands

There are two types of DOS command:

- Internal Commands
- External Commands.

DOS **internal commands** are stored in the COMMAND.COM file. These are loaded automatically into the memory during booting process. These include the commonly used commands. Internal commands are part of Command.com thus you see their names in directory listing. These commands remain in memory during the execution of DOS. Some of the internal DOS commands are: CLS, DIR, DEL, DATE, TIME, EXIT etc.

DOS **external commands** are those, which need special files for their execution. The DOS commands which are not frequently used are given as external commands. The three types of files that can run as an external commands are COM, EXE and BAT files. Some of the external DOS commands are: CHKDSK, DELTREE, FORMAT, XCOP etc.

How does DOS Organize Data?

DOS divides the secondary storage of the computer into logical areas called drives and each drive is recognized by a letter symbol followed by a colon. For example, the one floppy disk on your computer may be called **A:** and second floppy disk may be called **B:** drive. The hard disk may be assigned the letter C and is called **C:** drive and the compact disk may be called the **D:** drive. It is very common that the user may have a very large hard disk and divides it into logical parts called partitions. Each partition is also assigned a drive letter. For example, if you have a 20 GB hard disk, then you may use the **FDISK** utility to divide it into four partitions of equal size. In this case your floppy disk will be called the A: drive and four partitions of your hard disk may be called C:, **D:**, **E:**, and **F:** drives.

The data on each drive is organized in the form of directories and files.. For example, you may have four subdirectories named **DOS, Command, Windows** and **Data** on drive C: of your computer. You may create more directories on this drive or within the existing subdirectories. You can save data in the form of files on drive C: or in any of the directory on C:

At any given time DOS marks on of the directory as the working area and executes all commands with respect to that directory until specified explicitly. Such a directory is called the **present working directory.**

The most frequently DOS commands are as follows:

7.6.1   CD (Change Directory)

CD the most frequently used DOS command. This command is used to change the present working directory.

Suppose that you have a game PACEM installed on my computer on drive C: and in the directory GAMES. Then all the files of this game are in C:\GAMES\PACEM\ directory. This is called the path of these files. The path actually tells the logical location of the files on the disk. The PACEM.EXE file is also located in this directory and you need to load this file into memory and execute it to play the game. Before you start playing this game it is always a good idea to make C:\GAMES\PACEM\ directory as your present working directory as some executable files use the present working directory for different purposes.

Assume that the present working directory is C:\DOS. To change the present working directory from C:\DOS> to C:\Games\Pacem you can enter the following sequence of commands on the prompt.

<div align="center">

C:\>CD ..

C:\>CD GAMES

C:\>CD PACEM

</div>

Or we can type the single command CD C:\Games\Pacem.

Note:	In this command we have specified the full path of the directory.

What happens if a directory or path you have specified within your CD command does not exist? The answer is bad command or invalid directory.

Following examples show the use of this command:

Example 1: To change the directory to the parent directory you can use the following command.

C:\GAMES\PARA

This means that if your present working directory is **C:\GAMES\PARA**, then you can use **CD..** to change the present working directory to **C:\GAMES>**

Now suppose that the present working directory is **C:\GAMES\PARA** and you enter the following command:

CD D:\TEST** , your current directory will become **Test in **D:** drive.

Example 2: To change the present working directory from C:\GAMES\PACEM to D:\TEMP you can enter the following sequence of commands at the prompt.

<div align="center">

D:

CD TEMP

</div>

Note that to change the present drive from C: to D: you need to enter the drive name followed by colon at the command prompt.

Example 3: To change the present working directory to the root-drive you can use the following command:

**CD **

So if the present working directory is C:\GAMES\PACEM, then to change it to C:\ you can use the above command i.e. CD \

Example 4: Suppose the present working directory is C:\GAMES\ and this directory contains a subdirectory PACEM, then to change the present working directory to C:\GAMES\PACEM you can enter the following command at the prompt.

CD PACEM

Note that in this case you do not need to specify the full path of the directory.

Example 5: To change the present working directory from C:\GAMES to C:\GAMES\ PACEM\TEMP\ you can enter the following command.

CD PACEM\TEMP

The general syntax of Change Dirctory (CD) command is **CD[DRIVE:]PATH**

If we do not specify the drive name before the path, then path is taken to be from the present working directory.

Now that you know how to move in different directories, we can learn other important commands.

7.6.2 MD/(MKDIR)

This command is used to create subdirectories e.g. if the present working directory on your computer is C:\GAMES, then you can enter the following command to create a directory named NEWGAME within C:\GAMES directory.

MD NEWGAME

Now if you want to create another directory within NEWGAME subdirectory, then there are two possibilities.

You can use CD command to change the present working directory to NEWGAME and then use MD to create a subdirectory or you can create the directory by giving the command

MD NEWGAME\TEMP

In this case we do not need to change the present working directory. You can also give following command to make a new directory named **TEST** within **D:\TEST\TEMP**

MD D:\Test\Temp\Test

7.6.3   RD /RMDIR (Remove Directory)

If we have an empty directory and wand to delete it RD can be used the most general syntax of writing this command is :

RD [DRIVE][PATH] or RMDIR [DRIVE][PATH]

Example: To delete the directory TEMP from the directory C:\DATA you can enter the following command on the prompt

RD C:\DATA\TEMP
or
RD TEMP if the present working directory is C:\DATA

RD removes empty directories only. If a directory contains a file or a subdirectory, then RD does not delete this directory.

7.6.4   DIR

This is one of the most frequently used DOS command. It is used to display a list of the files and subdirectories that are in the directory. For example, to see the subdirectories and files in the present directory we can enter the following command on the command prompt.

DIR

The output of this command is shown in the adjoining figure.

Note that the command was executed from the directory **C:\Data** and it displays the following important information about this directory.

- The disk's volume label and serial number and the directory information.
- One directory or filename per line, including the extension.
- The file size in bytes.
- The date and time the file was last modified.
- The total number of files listed and their cumulative size.
- The total number of subdirectories listed.
- The free space (in bytes) remaining on the disk.

You can view the directory entries of any directory by using this command. For example, to view the contents of the directory **C:\DATA\PACEM** on your computer you can use the command

DIR C:\DATA\PACEM

So the general syntax of the **DIR** command is

DIR [DRIVE:][PATH]

It is very important to note that if the drive name is not specified, then the path is taken from the present working directory.

If the present working directory is **C:\DATA** and you enter the command **DIR PAPERS\FINAL** at the prompt, then the contents of the directory **C:\DATA\ PAPERS\FINAL** will be shown and contents of the directory **C:\PAPERS\FINAL** will not be shown.

You can also use the **DIR** command to see the existence and properties of a file in a directory. For example, to see the properties of the file **ABC.TXT** in the directory **C:\Data\Backup** you can use the following command at the command prompt.

DIR C:\Data\Backup\ABC.TXT

If the file and path of the files both exits, then the **DIR** command will display its properties on the screen.

Using Wildcards with DIR

Very often we want to search for a certain file based on some known property of the file name. For example, we may want to list all executable (. EXE) files in a directory or we may want to see only those files which have name starting with **car.** To list such files we can use wildcard characters * and ? are used as wildcard characters and have the following meanings:

- * is used to denote any number of characters.
- ? is used to denote exactly one missing character.

Example 1: Write a DIR command to show all files having .EXE as their extension in the directory DATA\GAMES\PACEM on the C: drive of your computer

Solution: As we are looking for all files having extension .EXE and the file can have any name so we can use the following command,

DIR C:\DATA\ GAMES\PACEM *.EXE

In this command the drive-name is C: the directory is \DATA\ GAMES\PACEM \ and the file name is * and extension is .EXE. As mentioned earlier the * means any number of characters so all the files having extension .EXE will be shown.

Example 2: To view all files in the present working directory which have name starting with **CE** you can use the following command.

DIR CE*.DAT

This command tells the DOS to show all files having names starting with the characters **CE** followed by any characters and then a . and then the extension **DAT**

Example 3: To view all files in the present working directory which have name containing the characters "**CE**" and having the character **A** in its extension you can use the following command.

DIR *CE*.*A*

This command tells DOS to show all file names starting with any character followed by the characters **CE** followed by any characters and then a dot (.) followed by any characters followed by the character **A** followed by any character

Example 4: To view all files in the present working directory which have name ending with character **X** you can use the following command.

DIR *X.*

Example 5: To view all files, in the present working directory, which have a four-character name you can use the following DOS command.

DIR ????.*

Note that one ? denotes one arbitrary character so ???? means four characters.

Example 6: To view all files, in the present working directory, which have a **X** as third character of the file name the following command can be used.

DIR ??X*.*

Example 7: To view all files, in the present working directory, which have a **X** as third last character of the file name you can use the following command.

DIR *X??.*

We can similarly use these wildcard characters to write many useful commands.

From the above discussion, it is obvious that the most general syntax of the **DIR** command is

DIR [DRIVE:][PATH] [FILENAME]

where

[DRIVE:][PATH]   specifies the drive and directory for which we want to see a listing and

[FILENAME]   specifies a particular file or group of files.

Using Switches with DIR

Often the directories contain many files and the **DIR** command cannot show the directory information on one screen. In such cases it is very hard to view the directory information as the directory information scrolls up and we do not get enough time to view the files. In such situations we can enter the following version of DIR command on the prompt.

DIR / P

The **/P** is called a switch and it makes the **DIR** command to display it's output one screen at a time and display the message **Press any key to continue**. So the user can view the information one full screen at a time by using the keyboard.

The switches used with the **DIR** command are given in the following table:

Switch	Purpose
/P	Displays one screen of directory list at a time.
/W	Displays the listing in wide format
/A[[:] [attributes]	Displays only the names of those directories and files with the attributes you specify. If you do not use this switch, DIR displays the names of all files except hidden and system files. If we use this switch without specifying any attributes, DIR displays the names of all files, including hidden and system files. We can specify the following Attributes of the file A. Files having Archive variable set are shown H Only hidden files are shown R Read only files are shown by using this option S System files are shown using this option
/O[[:] [sortorder]	Controls the order in which DIR sorts and displays directory names and filenames. If this switch is not used DIR displays the names in the order in which they occur in the directory. We can use this switch without specifying a sort order to sort the output in alphabetic order We can also specify other sort orders of our choice. Following is the list of choices for sorting the output of IR command N In alphabetic order by name

Switch	Purpose
	-N In reverse alphabetic order by name (Z through A) E In alphabetic order by extension -E In reverse alphabetic order by extension (Z through A) D By date and time, earliest first -D By date and time, latest first S By size, smallest first -S By size, largest first G With directories grouped before files -G With directories grouped after files
/S	Lists every occurrence of a file or directory in the specified directory and all of its subdirectories. This switch can be used to search a file.
/B	Lists each directory name or filename, one per line. This switch displays no heading information and no summary.
/L	Displays unsorted directory names and filenames in lowercase.

Example 1: To list all hidden files in the directory **C:\TEMP** use the following **DIR** command .

DIR C:\TEMP / AH

Example 2: To list all the files, which are read-only and hidden following **DIR** command can be used.

DIR / ARH

Example 3: To view all files and directories including the hidden and system files and directories from the present working directory, enter the following command on the prompt.

DIR /A

Example 4: Suppose you want **DIR** to display one directory listing after another, until it has displayed the listing for every directory on the disk in the C drive. Suppose that you also want DIR to alphabetize each directory listing and display it in wide format, and pause after each screen. For such a display you can use the following command.

DIR C:\ /S/W/O/P

DIR lists the name of the root directory, the names of the subdirectories of the root directory, and the names of the files in the root directory (including extensions). Then DIR lists the subdirectory names and filenames in each subdirectory.

Example 5: To list only the subdirectories in the present working directory. For this we can enter the following command on the prompt.

DIR /AD

Example 6: To list only the files in the present working directory and not the subdirectories you can use the following command.

DIR /A-D

Example 7: Suppose you want to save output of the **DIR** command in a file named **DIR.DOC.** You can do it by using the following command:

DIR > DIR.DOC

       If **DIR.DOC** does not exist, MS-DOS creates it. This technique can be used with many DOS commands.

Example 8: To display a list of all the filenames with the **.TXT** extension in all directories on drive C we can type the following command:

DIR C:*.TXT /W/O/S/P

       DIR displays in wide format, an alphabetized list of the matching filenames in each directory and pauses, each time the screen fills, until you press a key to continue.

       Similarly we can use the DIR command and it's switches in many useful ways.

7.6.5   ATTRIB

       Before we learn this important command it is important to know the meaning of attribute of a file. In DOS a file is one or more of the following type.

Read only file : The user can read such a file but cannot modify the file

Hidden File : The usual DOS commands are not applied on these files. So if a user gives a command to show files in a directory then the hidden files are not shown.

Archived File : The files have been archived. This means a backup of these files have been taken.

System Files : These files are used by the operating system to perform different functions. Device drivers are example of such files.

       DOS uses four variables {A, R, S, and H} to assign four attributes to each file. If a file has been archived {a historical backup has been taken} then the variable **A** is not set. This indicates that the file does not need to be archived or it has not been modified since last archive operation. If the file is read only then the variable **R** is set. If the file is a system, then the variable **S** is set and if the file is hidden, then the variable **H** is set.

       The **ATTRIB** command is used to view/change the attributes of files. For example, if the present working directory is **C:\DATA,** then following command can be used to display attributes of all files in this directory

          ATTRIB   or   **ATTRIB *.***

The results of this command are shown in the figure below.

Output of ATTRIB command
Figure

In the above figure, the file **OS.DOC** is read only, the file **QUESTION.DOC** is Archive, hidden and read only and the file **Report. DOC** is read only.

To display attributes of all files in the directory **C:\Data\Pacem** you can use the following syntax of the **ATTRIB** command.

ATTRIB C:\Data\Pacem*.*

The general syntax of this command is **ATTRIB [Drive:][Path][FileName]**

You can also use this command to view the attributes of a single file. For example, to view the attributes of a file named **pacem.exe** located in the directory **C:\Data\Pacem** you can use the following command.

ATTRIB C:\Data\Pacem\pacem.exe

You can also use **ATTRIB** to change the attribute of a file. For example, to make a file **REPORT.DOC,** in the present working directory, read-only you can use the following command.

ATTRIB +R REPORT.DOC

Similarly to reset the read only attribute you can use the following command.

ATTRIB -R REPORT.DOC

Here +R and –R are called the switches of the command. The following table gives the switches used with **ATTRIB** command.

Switch	Purpose
+R	Sets the Read-Only file attribute.
-R	Clears the Read-Only file attribute.
+A	Sets the Archive file attribute.
-A	Clears the Archive file attribute.
+S	Sets the file as a System file.
-S	Clears the System file attribute.
+H	Sets the file as a Hidden file.
-H	Clears the Hidden file attribute.

You can also set or reset the attributes of all the files using a single command given below.

ATTRIB –R or **ATTRIB –R C:\DATA**

If a file is hidden or system, then **ATTRIB** will not change its attribute as it is necessary to change these attributes before changing any other attribute .

Another switch used with this command is **/S,** by using this switch we can processes files in the current directory and all of its subdirectories.

We can use these wildcards (as described in the DIR command) in the ATTRIB command. For example, the following command will set the read only attribute of all files having name starting with C .

ATTRIB +R C*.*

We can also set the attributes of directories but cannot use wildcard characters for naming directories.

7.6.6 DEL (Erase)

It is another useful but dangerous command provided by DOS. It is used to delete the files from the disk. Following examples show the use of this command.

Example 1: To delete the CAT.TMP file from the TEST directory on drive C we can use either of the following commands:

                    DEL C:\TEST\CAT.TMP
                    ERASE C:\TEST\CAT.TMP

The general syntax of this command is

                    DEL [DRIVE:][PATH]FILENAME [/P]
                    ERASE [DRIVE:][PATH]FILENAME [/P]

where

[DRIVE:][PATH] specifies the drive and directory for which we want to see a listing and
[FILENAME]   specifies a particular file or group of files.

When **DRIVE** is not specified, then the **PATH** is taken relative to the current working directory. When the **PATH** is not specified, then the file is searched and deleted from the present working directory.

It is important to note that wildcard characters, as described in the **DIR** command, can also be used to specify the file names but it is a very dangerous practice sometimes, so these characters should be used with care.

This command has only one switch /P. If we use the /P switch, DEL displays the name of a file and prompts with a message in the following format:

FILENAME, DELETE (Y/N)?

We can press **Y** to confirm the deletion and **N** to cancel the deletion. IF wildcard characters were used to specify more than one file the message is shown for the next **FILENAME.** We can also press the CRTL+C keys to stop **DEL** command.

Following examples show the use of this command.

Example 2: To delete a file named **ABC.DAT** from the present working directory you can use one of the following commands.

DEL ABC.DAT or ERASE ABC.DAT

We can delete more than one file by giving a single **DEL** command as shown in the following example.

Example 3: To delete all the files from the directory named **TEST** on drive **C**, you can use either of the following commands:

DEL C:\TEST   or   **DEL C:\TEST*.***

When this form of DEL is used DEL displays the following prompt:

All files in directory will be deleted! Are you sure (Y/N)?

We can press **Y** and then **ENTER** to delete all files in the current directory, or press **N** and then **ENTER** to cancel the deletion. The following simple command can delete all files from the present working directory

DEL *.*

It is a very common error that users unintentionally delete all files so this command must be used with care. It is also important to note that this command can not be used to delete directories and if we try to delete a directory using this command all files within that directory are deleted but the directory itself is not deleted.

7.6.7 COPY

This is another very useful command provided by DOS. It is used to make copies of existing files. This command can also be used to combine multiple files into a target file. Following examples show the use of this command:

Example 1: The following command copies a file **MEMO.DOC** as **LETTER.DOC**. Both the files are in the current working directory

COPY MEMO.DOC LETTER.DOC

Example 2: To copy the **NOTE.TXT** file from the present working directory to the directory **C:\DATA** you can use the following command

COPY NOTE.TXT C:\DATA\NOTES.TXT

If the destination directory already contains a file named **NOTE.TXT** then DOS shows the following message.

OVERWRITE C:\DATA\NOTE.TXT: (YES/NO/ALL)

You can press **Y** and **Enter** to overwrite, **N** and **Enter** to cancel and **A** and **Enter** to overwrite all files. The last option is useful when more than one file are being copied.

You can use wildcard characters to specify more than one file to be copied. This is shown in the following example.

Example 3: To copy all files from **C:\DATA\PACEM** into the present working directory you can use the following command:

COPY C:\DATA\PACEM*.*

To copy data into the present working directory you do not need to specify the destination directory in the COPY command.

Example 4: To copy all files from the present working directory into the directory **C:\DATA \BACKUP** you can use the following command:

COPY *.* C:\DATA \BACKUP

If the destination directory already contains some files with the same name, then DOS shows the following message the overwrite message :

OVERWRITE C:\DATA\FileName :(YES/NO/ALL)

You can press **Y** and **Enter** to overwrite, **N** and **Enter** to cancel and **A** and **Enter** to overwrite all files.

Example 5: You can combine several files into one file by using the following **COPY** command.

COPY ONE.TXT + TWO.TXT + THREE.TXT FOUR.NEW

This command will create a new file named **FOUR.NEW** by combining the existing files **ONE.TXT**, **TWO.TXT** and **THREE.TXT** into a new file **FOUR.NEW** we can give the following command

You can also use wildcard characters to specify multiple files to be combined. For example to combine all **TXT** files in the present working directory into a new file named **UPDATE.TXT** you can use the following command:

COPY *.TXT UPDATE.TXT

It is important to note that if we do not specify a destination file, MS-DOS creates a copy with the same name in the present working directory. If the file to be copied is also in the present working directory then the COPY command stops and MS-DOS displays the following error message:

> **File cannot be copied onto itself**
> **0 File(s) copied**

7.6.8   DATE

This command is used to display the date and it prompts you to change the date if necessary.

The syntax of the Date command is **DATE [mm-dd-yy]**

This command sets the date we specify. Values for day, month, and year must be separated by periods (.), hyphens (-), or slash marks (/). It is important to note the following limitations e.g.

mm	1	through	12				
dd	1	through	31				
yy	80	through	99	or	1980	through	2099

IF you do not specify the date the DOS asks for the date. If the present date of your system is correct you can simply press the **Enter** key without entering the new date.

7.6.9   TIME

To display the current time or to display a prompt, by which you can change the current time enter the following command at the command prompt:

TIME

If you specify the time in an invalid format, MS-DOS displays the following message.

> **Invalid time**
> **Enter new time:_**

To keep the present time you can simply press the enter key

7.6.10   VER

This command is used to display the MS-DOS version number. To view the version number of MS-DOS on your system enter this command at the prompt:

VER

7.6.11   TYPE

This command is used to displays the contents of a text file on the screen. The original file is not modified when we use this command.

The syntax of this command is

TYPE [DRIVE:][PATH]FILENAME

If we display a binary file (e.g. executable files) using **TYPE** command we will see strange characters on the screen.

To display the contents of a file named **WORK.TXT** we can use the following command:

TYPE WORK.TXT

If this file does not fit on one screen we can use the MORE switch to see the file one screen at a time as follows:

TYPE WORK.TXT / MORE

7.6.12 FORMAT

The FORMAT command creates a new root directory and file allocation table for the disk. It can also check for bad areas on the disk, and it can delete all data on the disk. In order for MS-DOS to be able to use a new disk the disk must be formatted.

Following examples show the use of this command.

Example 1: To format a new floppy disk, in drive A using the default size, type the following command:

FORMAT A:

Example 2: To perform a quick format on a previously formatted disk in drive A, type the following command:

FORMAT A: /Q

Example 3: To format a floppy disk in drive A, completely deleting all data on the disk, type the following command:

FORMAT A: /U

This is commonly known as the unconditional format.

Example 4: To format a floppy disk in drive A and assign to it the volume label "DATA", type the following command:

FORMAT A: /V:DATA

After formatting a floppy disk, FORMAT displays the following message:

Volume label (11 characters, ENTER for none)?

The volume label can be a maximum of 11 characters (including spaces). If you do not want your disk to have a volume label, press ENTER.

When you use the FORMAT command to format a hard disk drive C:\ MS-DOS displays the following message:

WARNING, ALL DATA ON NON-REMOVABLE DISK
DRIVE C: WILL BE LOST!
Proceed with Format (Y/N)?_

To format the hard disk, press **Y**; if you do not want to format the disk, press **N**.

Note that it is very dangerous command and you may loose important data if you use this command carelessly.

Following table lists few switches commonly used with the **FORMAT** command

Switch	Purpose
/S	Copies the operating system files IO.SYS, MSDOS.SYS, and COMMAND.COM to a newly formatted disk so that it can be used as a system disk.
/Q	Specifies a quick format of a disk. With this switch, FORMAT deletes the file allocation table (FAT) and the root directory of a previously formatted disk, but does not scan the disk for bad areas. This switch is used to format only previously formatted disks that are in good condition.
/V:lbl	Specifies the volume label. A volume label identifies the disk and can be a maximum of 11 characters. If this switch is omitted or used without specifying a volume label, MS-DOS prompts for the volume label after the formatting is completed.
/U	Specifies an unconditional format of a disk. Unconditional formatting destroys all existing data on a disk and prevents from using the UNFORMAT command..

7.6.13 DISKCOPY

This command is used to make copy of the floppy disk and is not used for hard disk. Its syntax is **DISKCOPY Source Destination.**

For example the following command copies the disk in drive **A:** onto a disk in drive **B:**

DISKCOPY A: B:

The DISKCOPY command prompts you to insert the source and destination disks and waits for you to press any key before continuing.

After copying, DISKCOPY displays the following message:

Copy another diskette (Y/N)?

If you press **Y**, DISKCOPY prompts you to insert source and destination disks for the next copy operation. To stop the DISKCOPY process, press **N**.

You can use a single drive as both the source and destination drive. For example the following command copies a disk in drive **A:** onto a disk that will also be inserted in the same drive

DISKCOPY A: A:

DISKCOPY stores an image of the source disk and then copies it after you put the destination disk in the drive. If your computer does not have enough space in the memory or on the hard disk, then you may have to swap the source and destination floppies several times.

7.6.14 EDIT

Edit is the name of an executable file (**EDIT.COM**). When you enter Edit on the prompt DOS searches for this file and loads it into memory and starts its execution. This program is a text editor that can be used to create and edit ASCII text files. Following screen shows the Edit screen shown by this editor.
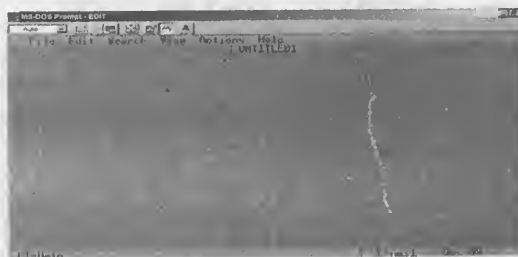
MS-DOS Editor allows you to **create, edit, save,** and **print ASCII** text files. Following is the syntax for entering this command.

EDIT [DRIVE:][PATH][FILENAME]

If you do not specify a file name then this editor starts a blank file.

7.6.15 SYS

This command is used to transfer the hidden MS-DOS system files (IO.SYS and MSDOS.SYS), the MS-DOS command interpreter (COMMAND.COM), to the disk specified in the command. When you transfer these files on a disk by using this command it can be used to start the system as it contains the DOS commands i.e. COMMAND.COM. The syntax of this command is

SYS [DRIVE1:][PATH] DRIVE2:

`where

[DRIVE1:][PATH] specifies the location of the system files. If you do not specify a path, MS-DOS searches the root directory on the current drive for the system files.

DRIVE2: specifies the drive to which you want to copy the system files.

The following example shows the use of this command:

Example: To copy the MS-DOS system files and command interpreter from the disk in the current drive to a disk in drive A, type the following command:

SYS A:

7.6.16 PROMPT

This command can be used to change the appearance of the command prompt. You can customize the command prompt to display any text you want,

Syntax of this command is **PROMPT [TEXT]** where **TEXT** specifies any text you want to show as the prompt.

The following table shows the some interesting character combinations you can use to form your prompt...

Symbol	Text	Symbol	Text	Symbol	Text
$Q	=	$$	$	$T	Current time
$D	Current date	$P	Current drive and path	$V	MS –DOS version number
$N	Current drive	$G	>	$L	<
$B	I (pipe)	$_	Line Feed		

When you use the PROMPT command without specifying a value for text, PROMPT resets the command prompt to the current drive letter followed by a greater-than sign (>). Following examples show the use of this command:

Example 1: To show my name followed by the > character on your prompt, you can use the following command.

PROMPT Mubasher Baig$g

Example 2: To show the present working directory followed by the > character you can use the following command.

PROMPT pg

Example 3: To display a two-line prompt in which the current time appears on the first line and the current date appears on the second line you can use the following command:

PROMPT TIME IS: T_DATE IS: $D

You can similarly use the symbols given in the above table to display other useful prompts.·

7.6.17 DELTREE

We know that the **RD (RMDIR)** command can not delete a directory if the directory is not empty. The DELTREE command can be used to delete the whole directory even if the directory is not empty. This command deletes a directory and all the files and subdirectories that are in it.

The syntax of this command is **DELTREE [/Y] [DRIVE:] PATH**

where **DRIVE: PATH** specifies the name of the directory you want to delete. You can specify more than one directory in a single command.

For example, to delete the directory C:\data\temp you can enter the following command on the prompt.

DELTREE C:\DATA\TEMP

When you use this command DELTREE shows you the following prompt:

Delete directory "C:\DATA\TEMP" and all its subdirectories [YN]

To delete the directory you can enter **Y** and to cancel you can enter **N**.

This command has only one switch **/Y** and when this switch is used DELTREE deletes the directory without first prompting you to confirm the deletion.

You can use wildcards with the DELTREE command, but use them with extreme care. If you specify a wildcard that matches with filenames then files will also be deleted. Following examples show the use of this command.

Example 1: To delete the **TEMP** directory on drive **C**, including all files and subdirectories of the TEMP directory, enter the following at the command at the prompt

DELTREE C:\TEMP

Example 2: To· delete the **TEMP** directory on drive C, and **TEMP1** directory on drive D, including all files and subdirectories of the TEMP and TEMP1 directory, type the following at the command prompt:

DELTREE C:\TEMP D:\TEMP1

Example 3: To All directories and files having **T** as the first name of their name you can use the following command.

DELTREE T*.*.

DELTREE prompts you before deleting any directory and file in this case. To avoid the prompt you can use the following command:

DELTREE /Y T*.*

7.6.18 XCOPY

This command is used to copy **directories**, their **subdirectories**, and files (except hidden and system files). We know that the copy command can not be used for copying the subdirectories so XCOPY command is very useful. Also this command has some useful switches for taking backup of the data. This command is provided in the form of an executable file. (**XCOPY.EXE**)

Following table lists some useful switches used with **XCOPY:**

Switches

Switch	Purpose
/Y	Indicates that you want XCOPY to replace existing file(s) without prompting you for confirmation.
/A	Used to Copy only source files that have their archive file attributes set.
/M	Used to Copy source files that have their archive file attributes set. Unlike the /A switch, the /M switch turns off archive file attributes in the files specified in source
/D: DATE	Used to Copy only those source files modified on or after the specified date.
/S	Used to copy non-empty directories and subdirectories
/E	Used to copy all directories and subdirectories even if the directories or subdirectories are empty.
/V	This switch is used to verify each file as it is written to the destination file to make sure that the destination files are identical to the source files.

Example 1: To copy all the files and subdirectories (including any empty subdirectories) from the present working directory into disk in drive A you can use the following command:

XCOPY *.* A: /S

Example 2: The following command uses the /D: and /V switches

XCOPY C:\Data\Backup: A: /D:01/18/04 /S /V

This   command   will   copy   only   those   files   from   the C:\ **DATA\BACKUP** directory onto the disk in drive A that were written on or after 01/18/93. After the files are written the XCOPY command compares the files to the source to make sure they are the same.

7.6.19 CHKDSK

This command is used to checks the status of a disk and to fix errors on the disk if any. This command can also be used to see if a certain file has any errors or is stored in contiguous blocks on the disk or not. This command has two switches /F and /V. /V switch is

used to display the files checked by CHKDSK and /F is used to correct error in a file or on the disk. It is important to note that all modern versions of DOS and Windows recommend the use of SCANDISK instead of CHKDSK as SCANDISK can detect and correct a much wider range of disk errors.

To use CHKDSK to check your drive C for errors enter the following command on the prompt.

CHKDSK C:

A sample output of this command when it is executed on a computer is given in the following figure:

```
Volume C    created 05-06-2003 10:04p
Volume Serial Number is 284C-1DD7
Errors found, F parameter not specified
Corrections will not be written to disk


    90 lost allocation units found in 3 chains.
    2,949,120 bytes disk space would be freed

2,146,500,608 bytes total disk space
    74,055,680 bytes in 635 hidden files
    33,128,448 bytes in 1,007 directories
1,291,321,344 bytes in 14,011 user files
    745,046,016 bytes available on disk

    32,768 bytes in each allocation unit
    65,505 total allocation units on disk
    22,736 available allocation units on disk

    655,360 total bytes memory
    615,104 bytes free

        Instead of using CHKDSK, try using SCANDISK.  SCANDISK can reliably detect and fix a much
wider range of disk problems.
```

To find out how much data is stored on drive C and how much space is still free, and to check the disk for errors, type the following command:

CHKDSK C:

7.6.20 PATH

This command is used to specify/view the directories MS-DOS should search for executable files. By default, the search path is the current directory only. This command has the following syntax:

PATH DRIVE: PATH; DRIVE: PATH...

To view the current search path you can use the following command:

PATH

To clear all search-path settings other than the default setting you can use the following command:

PATH;

It is important to note that MS-DOS always searches in the current directory first, before it searches directories in the search path.

You can specify more than one path for MS-DOS to search by separating entries with a semicolon (;)

For example, the following command specifies that MS-DOS is to search three directories to find commands

PATH C:\DOS\; C:\DATA\D:\GAMES; C:\COMMAND

7.6.21 VOL

This command displays the disk volume label and serial number, if they exist. The general syntax of this command is as follows:

VOL [Drive:]

Example:

VOL D:

This will display the volume of D: drive and its serial number

7.6.22 TREE

This command displays the folder structure of a drive or path. The general syntax of this command is:

TREE [Drive:] [Path] [/F] [/A]

/F   displays the name of files in each folder

/A   use ASCII instead of extended characters

Example:

TREE C:

This will display the folder structure of C: drive.

Exercise

1. Describe (a) System Software and (b) Application Software.
2. What is DOS? How is it different from Windows?
3. How many types of commands are available in DOS? Discuss briefly.
4. What is a language translator? Describe its types briefly.
5. What are switches and wild card? Discuss their uses in DOS with examples.
6. Define operating system. List important functions of operating system.
7. What is the difference between .com and .exe programs?
8. What is a directory, a volume label, and drive name?
9. How do you launch the command processor under Windows XP?
10. Describe the following terms:
    (i)   path
    (ii)   parent directory
    (iii)   subdirectory
11. How do you
    (i)   list all text files in subdirectory b:\reports\
    (ii)   list all files with name accounts under directory a:
12. Suppose you are working in directory C:\testdirectory. How whould you perform the following:
    (i)   create a new directory named **user**
    (ii)   change directory to C: by changing to parent directory twice
    (iii)   delete file named sample3.doc under the testdirectory
    (iv)   Remove the testdirectory after deleting files sample2.txt and sample3.doc.
13. Write DOS commands:
    (i)   To view current date

      (i)   change date to new date 2004-06-25

      (ii)   change date back to 2004-06-16 in one statement.

14. Take listing of: (a) all files under testdirectory when you are under C:\ (b) list all files of name sample under the testdirectory and (c) listing of all files of extension .doc under the testdirectory.

15. Write DOS commands to Erase

      (i)   sample.doc file under C:

      (ii)   sample4.doc file under testdirectory

      (iii)   All files under testdirectory.

16. Make testdirectory2 under subdirectory testdirectory when you are under C:\.

17. Explain the following commands:

      (i)   format

      (ii)   exit

      (iii)   find

      (iv)   pause

             print.

18. Change prompt to: (a) current time b) version number 8c) default drive (d) > character and (e) < character.

19. Write down the procedure for writing autoexec.bat file.

20. Explain sort and sys commands

21. Explain type, volume and xcopy commands.

22. Practice all important DOS commands using Windows Command Window.

23. Practice all important internal DOS commands using Windows Command Window.

24. Practice all important external DOS commands using Windows Command Window.

25. Discus in the class room the salient points of MS-DOS.

26. Discus in the class different DOS commands and their switches.

27. Fill in the blanks

      (i)   _____ and _____ convert a high level program into a machine program.

      (ii)   _____ provides the command line interface to the users.

      (iii)   **DIR /p** command is used to _____

      (iv)   _____ command can be used to delete all **.EXE** file from a directory.

      (v)   _____ is an external DOS command and _____ is an internal command

      (vi)   DOS stands for _____

      (vii) system software are necessary for the _____ use of computer

      (viii) _____ can be used to make a file read-only

      (ix)   _____ command can delete folders and all subfolders and files within it.

      (x)   The _____ switch is used with FORMAT command to conditionaly format a drive.

      (xi)   FDisk is a(n) _____ DOS command

28. **Match the following**

DIR	Interpreter
ATTRIB	Compiler
Operating system	View a directory
Line by line translation	Make a file read only
High level language	Memory management

29. **Choose the correct answer**

      (i)   XCOPY

            a. can copy subfolders as well

            b. is an external DOS command

            c. both a and b

            d. none of above

(ii) Windows
    a. has a GUI
    b. is not an operating system
    c. is a compiler
    d. all of above

(iii) DOS
    a. is an operating system
    b. is not an operating system
    c. has Graphical user interface
    d. none of above

(iv) Interpreter translate
    a. the assembly language program line by lin
    b. the source program line by line
    c. the source program as a whol
    d. none of the above

(v) dir ?lass.* command
    a. will list all files whose last four characters are **lass** with any extention
    b. will list all files starting with any character and with any extension
    c. will list all files whose name starts with any character but last four characters are **lass**
    d. none of the above

(vi) **rmdir** command
    a. makes directory **b.** removes directory **c.** copies file **d.** none of the above

(vii) **prompt** command
    a. confirms deletion of file **b.** changes prompt **c.** searches a directory

(vi)   d. none of the above

(vii) **dir *.***
    a. lists all files of extension .doc. **b.** lists all files
    c. lists all files of name sample. **d.** None of the above

30. **Mark the following as True/False**
(i) An Assembler converts a high level language program into machine language
(ii) MD command can be used to delete directories as well.
(iii) **XCOPY /s *.* d:** command copies all the files from the present directory to d:\copy\ directory
(iv) FORMAT a: command will delete all existing data from the a: drive and prepare it for storing data
(v) DOS external commands resides in Command.com
(vi) Diskcopy can not copy files from hard disk
(vii) Batch file contain multiple DOS commands to be executed
(viii) Time command can be used to change the current time
(ix) DELTREE is an internal DOS command
(x) FORMAT command can not format an already formatted disk

Answers

Q.27
(i) Compiler, Interpreter   (ii) DOS   (iii) List files and directories page wise iv. del *.exe
(v) DELTREE, DIR   (vi) Disk Operating System   (vii) effective   (viii) ATTRIB Command
(ix) DELTREE   (x) /u   (xi) External

Q.31
(i) c   (ii) a   (iii) a   (iv) b   (v) c
(vi) b   (vii) B   (viii) b

Q.32
(i) F   (ii) F   (iii) F   (iv) T   (v) F
(vi) T   (vii) T   (viii) T   (ix) F   (x) F